In the Specification

Delete the paragraph beginning on page 1, line 8, and replace it with the following paragraph:

The relevant prior art, upon which the present invention includes improvements to, is the *make* utility originally developed to run on UNIX® systems. A brief discussion of the *make* command or utility in a UNIX environment is herein discussed to serve as background in disclosing the present invention. The *make* utility is a software engineering tool for managing and maintaining computer programs. Typically, a program may ~~consists~~ consist of many component files, and as the number of files in the program increases, the compile time also increases. Similarly, the complexity of compilation commands can increase with the program complexity. In the same way, the likelihood of human error in executing many compilation commands and updating source files can also ~~increases~~ increase with the program complexity.

Delete the paragraph beginning on page 1, line 18, and replace it with the following paragraph:

The *make* utility uses a descriptor~~descripter~~ file containing dependency rules, macros, and suffix rules to instruct *make* to automatically rebuild the program whenever one of the program component files is modified. The *make* utility operates by following rules that are provided in its descriptor file, typically called *makefile*. The *make* utility saves compile time by selectively recompiling only the files that were effected by changes.

Thus, the *make* utility simplifies the problem of keeping track of modified files, recompiling files, and re-linking those files to produce an executable program.

Delete the paragraph beginning on page 3, line 28, and replace it with the following paragraph:

More specifically, *make* operates by comparing the date and time of a source file with the date and time of the associated object file. Together, the date and time are called the time stamp. If the comparison shows that the source file is newer ~~then~~ than the object file, or if the object file does not exist, *make* performs the task listed in the *make* file to convert the source file into an object file. In contrast, if the object file is newer then the source file, then *make* recognizes that is does not have to recompile the source file.

Delete the paragraph beginning on page 4, line 6, and replace it with the following paragraph:

The *make* utility requires that a target/dependency line must begin in the first column of the line. A command line must be indented. A target is a name followed by a colon. The name appears at the beginning of the line. Two types of dependencies occur in a *make* file, direct dependencies and indirect dependencies. A direct dependency appears with the target. This means that the target depends on the file or files listed after the colon. An indirect dependency is indirectly related to the first target. However, typically, multiple *makefiles* are ~~need~~ needed to specify cross-directory file dependencies.